

# Home

FEST is a collection of open source APIs that aim at simplifying software testing. It is composed of various modules. All of them can be used with [TestNG](#) or [JUnit](#).

## Functional Swing GUI Testing

This module provides a simple and intuitive API for functional testing of Swing user interfaces, resulting in tests that are compact, easy to write, and read like a specification. Tests written using FEST-Swing are also robust. FEST simulates actual user gestures at the operating system level, ensuring that the application will behave correctly in front of the user. It also provides a reliable mechanism for GUI component lookup that ensures that changes in the GUI's layout or look-and-feel will not break your tests.

FEST makes [troubleshooting failures](#) a lot easier. It can take screenshots of the desktop at the moment of a test failure, to make it easier to determine if the failure was due to a programming error or an unexpected environment event. When a GUI test fails, FEST also provides useful information in the failure message, such as a [nicely-formatted GUI component hierarchy](#), allowing developers to quickly inspect the cause of the failure.

The following example simulates a user logging-in into a Swing application. The test verifies that an error message is displayed if the user forgets to enter her password.

```
dialog.comboBox("domain").select("Users");

dialog.textBox("username").enterText("alex.r
uiz");
    dialog.button("ok").click();
    dialog.optionPane().requireErrorMessage()
                        .requireMessage("Please
enter your password");
```

FEST-Swing can test desktop applications as well as applets (in a viewer and in-browser.)

For more details, please visit the [Swing Module Wiki](#).

## Fluent Assertions

This module provides a [fluent interface](#) for assertions. FEST's assertions are incredibly easy to write: just type "assertThat" followed the actual value and a dot, and any Java IDE will show you all the assertions available for the type of the given object to verify. No more confusion about the order of the "expected" and "actual" values. Our assertions are very readable as well: they read very close to plain English, making it easier for non-technical people to read test code.

Here are some examples:

```
int removed = employees.removeFired();
    assertThat(removed).isZero();

    List<Employee> newEmployees =
employees.hired(TODAY);
    assertThat(newEmployees).hasSize(6)
                                .contains(frodo,
sam);

    assertThat(yoda).assertInstanceOf(Jedi.class)
                    .isEqualTo(foundJedi)
                    .isNotEqualTo(foundSith);
```

For more details, please visit the [Assertions Module Wiki](#).

## Fluent Reflection

This module provides an intuitive, compact and type-safe [fluent API](#) for Java reflection. It makes Java reflection tremendously easy to use: no more casting, checked exceptions, `PrivilegedAction` or calls to `setAccessible`. FEST's reflection module can even overcome the limitations of generics and type erasure.

```
Person person =
constructor().withParameterTypes(String.class)

.in(Person.class)

.newInstance("Yoda");

method("setName").withParameterTypes(String.class)

.in(person)
.invoke("Luke");
```

For more details, please visit the [Reflection Module Wiki](#).

### EasyMock Template

Our EasyMock template eliminates code duplication (repetitive calls to replay and verify) and clearly separates expectations from code to test.

```
@Test public void shouldUpdateEmployee() {
    new EasyMockTemplate(mockEmployeeDao) {
        protected void expectations() {
            mockEmployeeDAO.update(employee);
        }

        protected void codeToTest() {
            employeeBO.updateEmployee(employee);
        }
    }.run();
}
```

**Note:** We are no longer developing the EasyMock template. We are currently replacing EasyMock with [Mockito](#) in our own tests. Since, in our opinion, Mockito's API is really nice and compact, it does not require a template.

If you are an EasyMock user, please visit the [EasyMock Template Wiki](#).